Software Requirements Specification

for

BookItNow

Version 1.0

Prepared by

Group Name: BookItNow

Utkarsh Verma	60004210205	theutkarsh.verma00@gmail.com
Aditya Dighe	60004210212	ardighe7@gmail.com
Sean Fargose	60004210224	fargose.sean2808@gmail.com

Instructor: Dr. Kiran Bhowmick

Course: Software Engineering

Lab Section: C31

Teaching Assistant: Dr. Kiran Bhowmick

Date: 7th February 2024

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Aditya Dighe Utkarsh Verma	This version of software requirements specification is a detailed view of our product BookItNow, which is an online event and show	11/02/24

Version	Primary Author(s)	Description of Version	Date Completed
	Sean Fargose	booking application. This is the first draft of the product perspective, its functionalities, the users, the operating systems, the environments etc.	

1 Introduction

1.1 Document Purpose

This Software Requirements Specification (SRS) defines the capabilities and intended behavior of BookItNow, an event booking application. It serves as a comprehensive description of BookItNow's functionality and constraints to be used for development and testing purposes.

The SRS outlines what the system should do from the standpoint of an end-user, specifying all the use cases and interactions involved in booking events. It provides a baseline for the development team to design and build the system to client expectations, while meeting all technical and business requirements. The detailed functional and non-functional requirements covered in this SRS will serve as the guidelines for the Instructor and development team to evaluate the completed system.

1.2 Product Scope

BookItNow is an online event booking application that allows users to securely reserve tickets for movies, concerts, sports events, and more. The key goal of BookItNow is to provide a seamless, integrated platform for booking event tickets and related purchases like accommodation and food delivery.

Key benefits offered by BookItNow include a streamlined booking process, personalized recommendations, bundled travel/event packages, and in-seat food ordering. This simplifies event attendance planning for users. The overarching objectives are: 1) Enhance user convenience by centralizing bookings 2) Drive revenue growth through targeted upsells and packages 3) Collect user data to enable personalized recommendations. By improving the booking experience, BookItNow aims to establish itself as the preferred online destination for end-to-end live entertainment planning.

1.3 Intended Audience and Document Overview

The key intended readers of this SRS document are the client who commissioned the BookItNow system, the professor who assigned the project, and the developers building the software.

The SRS begins with an introduction providing background context and defining key terms. The Overall Description and Specific Requirements sections provide a comprehensive overview of all system functionality and constraints. These sections are most relevant to understand overall scope.

Later sections dive into finer details around external/internal interfaces, functional/non-functional requirements, and supplementary content like the data dictionary. These granular technical sections will be most useful for developers.

The document structure flows from general to more niche system information. It's suggested clients and professors focus on the early sections for big picture vision, while developers concentrate efforts on technical specifications in the later portions. The appendices contain complementary information as quick references.

1.4 Definitions, Acronyms and Abbreviations

API - Application Programming Interface

SMS - Short Message Service

SRS - Software Requirements Specification

UI - User Interface

UML - Unified Modeling Language

1.5 Document Conventions

The formatting and typographical conventions followed in this SRS document:

Formatting Conventions

- Font: Arial, Size 11
- Single spaced paragraphs
- Margins: 1" on all sides
- Section and subsection titles formatted as per IEEE template
- Italics used to indicate developer comments and instructions

Naming Conventions

- Requirement IDs follow format REQ-001, REQ-002 for traceability
- Tables and figures numbered sequentially
- File names match SRS section names for easy cross-reference

Highlighting Conventions

- Key terminology is underlined in requirements for visibility
- TBD (To Be Determined) used to indicate requirements needing clarification
- TODO tags critical development tasks needing completion per design

Adhering to these IEEE formats and conventions aims to enhance document readability, organize content, and simplify SRS-based development.

1.6 References and Acknowledgments

The references used in this SRS document, formatted as per the IEEE citation style guide:

[1] R. Nicole, "User Interface Style Guide for BookItNow," BookItNow Inc., Toronto, ON, Canada, Style Guide v1.0, Jan. 2022.

[2] A. Singh, "Vision and Scope Document - BookItNow," BookItNow Inc., Toronto, ON, Canada, Scope Document v2.1, Mar. 2022.

[3] P. Jones et al., "System and Software Requirements Specification - MiniThermostat," Acme Solutions Inc., Towson, MD, USA, SRS Rev. 1.4, Feb. 2023.

[4] S. Williams, "Use Cases for Event Booking Applications," 2023. [Online]. Available: https://bookingapps.com/resources/use-cases

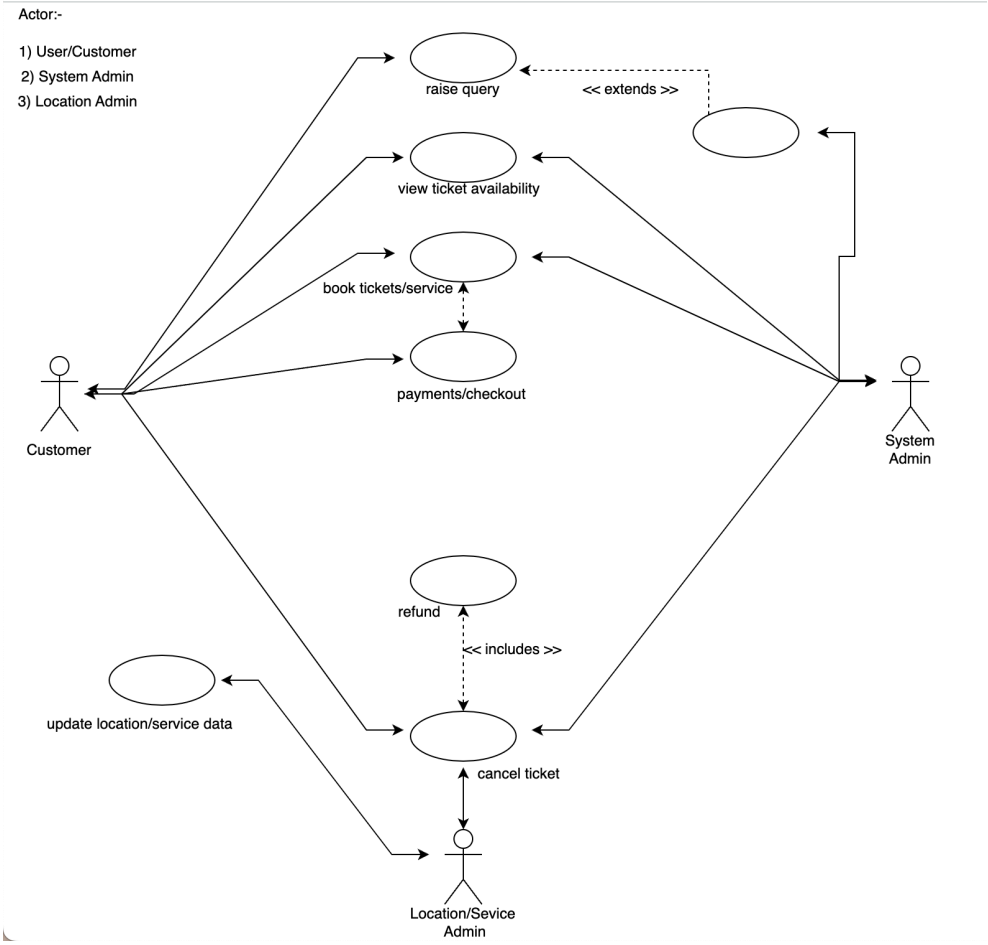
This references key supplementary project documents either produced internally or from external sources. They provide additional context, guidelines, and examples that guided the development of this SRS.

2 Overall Description

2.1 Product Perspective

BookItNow is a new online booking application developed from scratch to address limitations with existing entertainment ticketing platforms. It aims to provide an integrated one-stop solution for event bookings and related purchases, consolidating the fragmented processes on current vendor websites.

The high-level system architecture follows a modular client-server model as depicted:



The BookItNow front-end provides the user interface portal for customers to search, select, and purchase event tickets. This connects via API to the centralized BookItNow server and database which orchestrates bookings while integrating with external third-party systems for necessary event information, payments, analytics, and delivery fulfillment.

The product is intended as an independent greenfield development, albeit with the future scope to potentially enhance with additional modules for venue/artist management purposes.

2.2 Product Functionality

Here is a summary of the major functions of BookItNow:

Key Functions:

- Search and browse available event listings based on criteria
- Select ticket quantities and seating options
- Purchase tickets by providing order and payment details
- Receive electronic ticket confirmation via email/SMS
- Order menu food and drinks for direct delivery to seat
- Book bundled event travel packages including flights, hotels
- View personalized recommendations of events based on past activity and preferences
- Provide reviews and ratings for events attended
- Manage user account details and booked orders

This high-level data flow diagram depicts the key use cases and flow of information between the users and the system components that support each major function of booking, ordering, reviews, and account management. Additional details on the sub-processes and data models will be covered in subsequent sections. Please let me know if you need any clarification or expansion on this overview diagram.

2.3 Users and Characteristics

This is a description of the BookItNow users and their characteristics:

General Public

- Use sporadically for booking tickets for self and friends/family
- Varied technical expertise and experience with ticket booking sites
- Key functions: Browse events, purchase tickets & food, read reviews

Frequent Event-goers

- Skilled at using booking sites and repeat customers
- Rely on quick and smooth ticket purchasing experience
- Additional usage of travel packages and recommendations

Businesses and Resellers

- Purchase tickets in bulk for resale or promotion
- Require batch booking options and order tracking/metrics
- Privileged backend access for managing large orders

The general public makes up the largest segment of end-users. Ensuring a streamlined booking process for casual, infrequent users is most critical.

Frequent event-goers are the second priority as repeat engaged customers. Their complex usage and loyalty contributes significantly to revenue.

Resellers and businesses have niche batch order needs. While important, their complex functionality does not impact wider system adoption.

2.4 Operating Environment

This is the description of the operating environment for BookItNow:

BookItNow will operate as a web application compatible with common desktop and mobile browsers like Chrome, Firefox, Safari and operate on a client-server architecture. The front-end UI will be accessible from Windows 10+, macOS 10.14+, iOS 12+ and Android 8+ devices without needing any specialized software installation.

The back-end will run on Linux servers hosted securely on cloud infrastructure to interface with databases, payment gateways, ticketing providers and other integrated platforms throughPUBL API protocols. The system is designed for high availability with adequate redundancy, scalability and cybersecurity controls.

At a minimum, BookItNow requires the servers to run Linux, MySQL, SSL certificates and dedicated connectivity along with sufficient compute for application and database layers. The UI will support responsive design accessible from both desktop computers and mobile devices with latest OS versions.

2.5 Design and Implementation Constraints

The potential design and implementation constraints for BookItNow:

1. Integration with third-party payment gateways limits available options for processing financial transactions to compatible solutions having public APIs.
2. The use of SMS-based ticket confirmation constraints messaging to templates provided by SMS providers.
3. Branding and style guidelines restrict the UI design options to maintain consistent look and feel.
4. Collecting user activity data for recommendations must respect privacy laws and obtain consent.
5. Peak event bookings supporting heavy traffic will necessitate scaling cloud infrastructure to handle load.

Some additional constraints could include:

6. Delivering a minimum viable product within tight deadlines.
7. Legacy database technologies constrain options for new data persistence layer.
8. Real-time food delivery logistics restricts technology choices.

2.6 User Documentation

Description of the user documentation to be provided with BookItNow:

The user documentation for BookItNow will include in-app guidance, FAQs, and online support resources to assist end-users. Contextual help displayed on screens and interactive walkthrough overlays will illustrate core booking flows. Frequently asked questions around account setup, payments, refunds etc. will address common points of confusion. Comprehensive support articles and tutorial videos to be hosted on the BookItNow website and knowledge base will build user proficiency. The online help content will educate end-users on all functionality while minimizing friction during the purchase process, enabling self-sufficient adoption of the platform.

2.7 Assumptions and Dependencies

Some of the key assumptions for the BookItNow platform that could impact requirements:

- Third-party payment gateways will have 99.95% uptime and sufficient transaction throughput to handle peak event bookings. Any issues could severely impact sales.
- Food fulfillment partners will accurately deliver 80% of items ordered to the correct event location and seats. Delivery performance below expectations could negatively impact service.
- Users will provide honest and representative reviews for events attended at least 50% of the time. Inaccurate or skewed reviews affect recommendations quality.
- The average user will book tickets for 2 events per year and spend 3 hours browsing the platform monthly. Variations could alter expected platform load and metrics.

If the actual operating conditions differ materially from these assumptions around third-party dependencies, user behavior, or environmental factors, the requirements and platform architecture may need reassessment.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The key user interfaces that interact with end users of BookItNow are:

1. **Landing and Home Page**: Serves as the entry point with search bars to find events. Navigation links provide account login, signup options and access to static information pages.

2. **Event Listing Page**: Displays paginated search results for user queries showing event name, venue, date, availability etc. Allows filtering, sorting and selecting specific events.

3. **Event Details Page**: Provides complete event information including timing, venue maps, seating layouts and ticket classes. Integrates ticketing provider inventories for exact seat maps.

4. **Shopping Cart and Checkout Flow**: Enables users to select ticket quantities per type, add promotional codes, redeem gift cards and proceed to a multi-step checkout process collecting user, payment method and delivery details.

5. **Order Confirmation and Tickets Page**: Displays confirmed ticket purchases with QR/barcodes for entry along with options for reprints, downloads, sharing over email etc.

6. **User Account and Profile Page**: Offers dashboard overviews of order history, saved payment methods, newsletter preferences and personal details. Includes account management flows like login, password changes and adding payment methods.

7. **Help and Support Pages**: Provides self-help resources like FAQs, user forums and support request ticketing for assistance from customer representatives.

3.1.2 Hardware Interfaces

The hardware interfaces for BookItNow are:

Client Devices

The client-side application interfaces with end user devices like desktops, laptops, smartphones and tablets. Supported client platforms:

1. Mobile Operating Systems: Android, iOS
2. Desktop OS: Windows, MacOS
3. Minimum Screen Resolution: 1024x768
4. Input Devices: Touchscreen, Mouse, Keyboard

Server Environment

The backend system interfaces with:

1. Cloud platform services like AWS, Azure for hosting environment
2. Load balancers for traffic distribution to application servers
3. Firewalls and VPN Gateways for secure external access

Payment Systems

Payment processing involves interfaces with:

1. Payment gateway systems like Stripe, PayPal
2. Merchant bank networks like Visa, MasterCard
3. Bank core systems for payment settlement

3.1.3 Software Interfaces

The key software interfaces used within BookItNow are:

Operating System

BookItNow backend components for application, database, storage and more will run on Linux-based OS environments like Red Hat Enterprise Linux, CentOS to leverage optimized LAMP stack deployments.

Databases

MySQL 8.x is the default relational database, interfacing via SQL for data access and ORM frameworks. NoSQL caches may complement session data and static content.

Web and App Servers

Node.js 12.x+ runtime will support the JavaScript based application logic and REST API endpoints interfacing with client devices and external systems. Nginx will proxy requests to Node.js processes.

Payment Systems

Pre-certified libraries and SDKs will handle integrations with payment gateways like Stripe, PayPal, facilitating the capture and processing of transactions.

3.1.4 Communications Interfaces

The key communication protocols and interfaces used by BookItNow are:

Web Browser Interactions

BookItNow user interfaces for web and mobile apps communicate over HTTP/HTTPS following REST API patterns. TLS 1.2+ encryption secures all client-server interactions and data transfers conforming to industry standards.

Server-to-Server Communication

Back-end application and database server communications occur internally over secured VPN tunnels. Encryption mechanisms safeguard all data access and transmission conforming to protocols like AES-256 and TLS 1.2+.

Notifications and Alerts

External notifications leverage communication channels like:

1. Email with end-user encrypted SMTP tunnels for ticket dispatches.
2. SMS gateway providers for booking confirmations to mobile phones.
3. Native mobile push notifications over APNs and FCM.

3.2 Functional Requirements

3.2.1 Search and Browse

1. Allow searching event listings by keywords, location, date, category etc.
2. Support filter, sort and pagination on search results based on criteria like timing, venue, availability etc.
3. Provide location-based recommendations and personalized suggestions based on user history and preferences.
4. Display event images, videos, seating maps and integrated ratings from review sites.

3.2.2 Ticket Purchase and Delivery

1. Allow selection of ticket quantities per category and allocate best available seats.
2. Capture contact information, payment methods etc. through multi-step checkout flows.
3. Validate payment details with gateways, process charges and fees.
4. After payment capture, trigger ticket generation and dispatch confirmation email and SMS.
5. Allow reprinting tickets from user accounts along with cancellations based on event policies.

3.2.3 Travel and Add-ons

1. Recommend flights, hotels and local transportation to users based on event location and dates.
2. Provide integrated booking for travel products and bundled event packages.
3. Capture orders for food, beverages and merchandise for delivery at event venue.
4. Track status and timing for delivery confirmation messages for all bundled add-ons like meals, transport etc.

3.2.4 User Accounts and Administration

1. Account management capabilities for profile creation, payment settings, order history and more.
2. Administrative interfaces to handle customer support, content management and ticket inventory status.
3. Dashboard analytics on vital metrics like sales volumes, conversion rates, popular events etc.
4. Management capabilities for creating and tracking promotional offers.

3.3 Behaviour Requirements

3.3.1 Use Case View

Admin:

Likely referring to an internal system/application administrator user with privileged access to configure software settings, manage databases, view analytics, oversee operations.

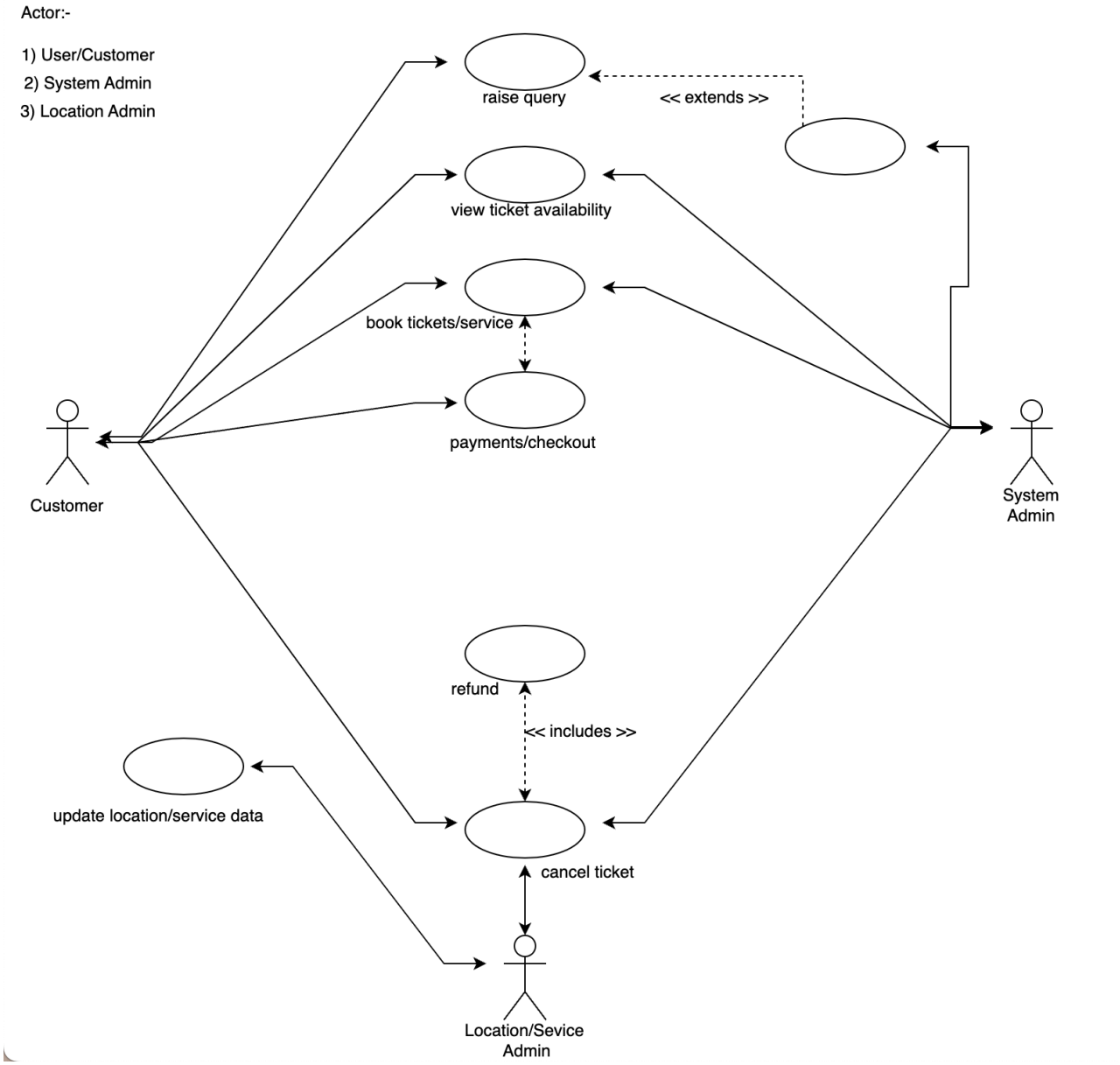
Customer:

An end-user customer that utilizes the primary services and capabilities offered by the software system under consideration. May interact through a front-end portal or client-side application.

Location Admin:

Potentially a third-party service provider that enables geo-location capabilities within the target system. Could control APIs, datasets, and dashboards related to maps, spatial analysis, logistics optimization leveraging location data.

Academic Year: 2023-24



4 Other Non-functional Requirements

4.1 Performance Requirements

The key performance requirements for BookItNow are:

1. Response time: The application should respond and load pages within 2 seconds for 90% of user requests, under peak load conditions of up to 10,000 concurrent active users.
2. Concurrent users: The system is expected to support 2,000 concurrently logged in users with acceptable response times as per the baseline established. This is projected to scale up to 10,000+ concurrent user sessions at peak usage times.
3. Transaction processing throughput: The booking transaction throughput must allow up to 200 ticket purchases per minute successfully with sub-second response times. Bulk order throughput from resellers should support 50 orders/minute.
4. Search function latency: Event search results must populate with a server response within 1.5 seconds when users apply filters based on criteria such as location, date, category etc. Cold searches without caches should return under 3 seconds.
5. System availability: BookItNow has a target monthly uptime SLA of 99.95% uptime allowing for planned maintenance downtimes. The ticketing system cannot remain unavailable for booking during any events or peak periods.

4.2 Safety and Security Requirements

The safety and security requirements of the BookItNow application are as follows:

1. Validation checks shall be implemented on all input forms to prevent faulty or incomplete orders from being processed.
2. Confirmation alerts shall be shown for all financial transactions. Email and SMS notifications shall serve as receipts.
3. All user-entered inputs shall be sanitized server-side to prevent script injection attacks or unauthorized data access attempts.
4. Passwords must be stored post hashing/salting to ensure leaked credentials cannot be used to compromise accounts.
5. SSL/TLS Encryption shall secure all connections between user devices and servers/databases.
6. Username and password based user authentication shall be used to verify identities against secure directory services.
7. Granular access controls with role-based permissions shall restrict functionality as per defined user roles.
8. Complete logs of user activities and system events shall be maintained for auditing and debugging.

9. Industry regulations like PCI DSS for payment processing and GDPR for data privacy shall be adhered to.
10. Fraud detection systems shall be employed to identify and prevent suspicious transactions or activities.

4.3 Software Quality Attributes

4.3.1 Reliability

Reliability of BookItNow shall be achieved through:

1. Multiple redundant servers and hot failover to ensure high availability.
2. Transaction atomicity, consistency, isolation, and durability (ACID) compliant database.
3. Automated unit, integration and performance testing across environments.
4. Monitoring dashboards tracking uptime, failovers, and system health metrics.
5. The production environment is targeted for 99.95% monthly uptime.

4.3.2 Usability

BookItNow shall prioritize usability by:

1. Conducting UI/UX testing and surveys to meet a user satisfaction score of 4/5.
2. Documenting interface standards and style guides for intuitive, consistent design.
3. Optimizing forms and flows for ease of use with tooltips and field descriptions.
4. Supporting self-service functionality by including tutorial videos and chatbots.

4.3.3 Maintainability

Maintainability will be ensured through:

1. Code modularity following SOLID design principles for improved extensibility.
2. Usage of leading frameworks like React and Node.js to ease developer onboarding.
3. Higher test coverage with reports and automated test suites for safety enhancing existing flows.
4. Version control disciplines and code reviews prior to migration across environments.

5 Other Requirements

*<This section is **Optional**. Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>*

Appendix A – Data Dictionary

<Data dictionary is used to track all the different variables, states and functional requirements that you described in your document. Make sure to include the complete list of all constants, state variables (and their possible states), inputs and outputs in a table. In the table, include the description of these items as well as all related operations and requirements.>

Appendix B - Group Log

<Please include here all the minutes from your group meetings, your group activities, and any other relevant information that will assist the Teaching Assistant to determine the effort put forth to produce this document>